

AD-A171 025

Technical Report 684

**Automated Author Aiding System Conference:
Final Report**

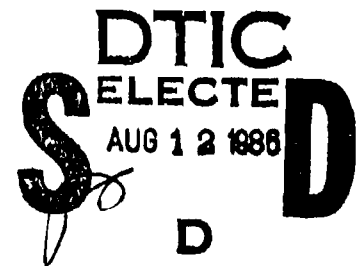
edited by

Nancy K. Atwood

University of California at Los Angeles

Jack H. Hiller

Army Research Institute



ARI Field Unit at Presidio of Monterey
Training Research Laboratory



U. S. Army

Research Institute for the Behavioral and Social Sciences

July 1985

Approved for public release; distribution unlimited.

DTIC FILE COPY

86 8 11 119

U. S. ARMY RESEARCH INSTITUTE FOR THE BEHAVIORAL AND SOCIAL SCIENCES

A Field Operating Agency under the Jurisdiction of the
Deputy Chief of Staff for Personnel

EDGAR M. JOHNSON
Technical Director

WM. DARRYL HENDERSON
COL, IN
Commanding

Research accomplished under contract for
the Department of the Army

Jet Propulsion Laboratory, University of California at Los Angeles

Technical review by

Joseph Psotka
Robert J. Seidel

NOTICES

DISTRIBUTION: Primary distribution of this report has been made by ARI. Please address correspondence concerning distribution of reports to: U.S. Army Research Institute for the Behavioral and Social Sciences, ATTN: PERI-POT, 5001 Eisenhower Ave., Alexandria, Virginia 22333-5600.

FINAL DISPOSITION: This report may be destroyed when it is no longer needed. Please do not return it to the U.S. Army Research Institute for the Behavioral and Social Sciences.

NOTE: The findings in this report are not to be construed as an official Department of the Army position, unless so designated by other authorized documents.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER ARI Technical Report 684	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) AUTOMATED AUTHOR AIDING SYSTEM CONFERENCE: FINAL REPORT		5. TYPE OF REPORT & PERIOD COVERED Final Report June 84 through June 85
		6. PERFORMING ORG. REPORT NUMBER --
7. AUTHOR(s) Edited by Nancy K. Atwood, Center for the Study of Evaluation, University of California at Los Angeles (UCLA) and Jack H. Hiller, ARI Field Unit, Presidio of Monterey, CA		8. CONTRACT OR GRANT NUMBER(s) NAS7-918, Task Order No. 182/ 183
9. PERFORMING ORGANIZATION NAME AND ADDRESS Jet Propulsion Laboratory California Institute of Technology 4800 Oak Grove Drive, Pasadena, CA 91109		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS 20263743A794 4311 102 7725
11. CONTROLLING OFFICE NAME AND ADDRESS Army Research Institute 5001 Eisenhower Avenue Alexandria, VA 22333-5600		12. REPORT DATE July 1985
		13. NUMBER OF PAGES 75
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Army Research Institute Field Unit PO Box 5787 Presidio of Monterey, CA 93944-5011		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE --
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report) --		
18. SUPPLEMENTARY NOTES Contracting Officer's Representative, Jack H. Hiller		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Training Intelligent Computer Assisted Instruction Author Aiding Artificial Intelligence		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) This report includes the proceedings of the Automated Author Aiding Systems Conference held at UCLA on June 6-7, 1984. The purpose of the conference was to determine the current state of the art in computerized aids for authors by providing a forum for the exchange and synthesis of ideas. Individuals attending the conference included researchers from the Department of Defense and academic sector with relevant interests, representatives of Department of Defense agencies and other organizations with interests in the development and application of such systems, and representatives of vendors of author aiding systems.		

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

1 SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Technical Report 684

Automated Author Aiding System Conference: Final Report

edited by

Nancy K. Atwood

University of California at Los Angeles

Jack H. Hiller

Army Research Institute

ARI Field Unit at Presidio of Monterey

Jack Hiller, Chief

Training Research Laboratory

Harold F. O'Neill, Jr., Director

U.S. ARMY RESEARCH INSTITUTE FOR THE BEHAVIORAL AND SOCIAL SCIENCES

5001 Eisenhower Avenue, Alexandria, Virginia 22333-5600

Office, Deputy Chief of Staff for Personnel

Department of the Army

July 1985

Army Project Number
20263743A784

Education and Training

Approved for public release; distribution unlimited.

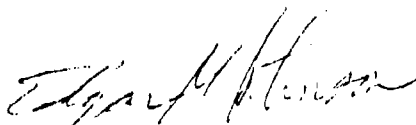
ARI Research Reports and Technical Reports are intended for sponsors of R&D tasks and for other research and military agencies. Any findings ready for implementation at the time of publication are presented in the last part of the Brief. Upon completion of a major phase of the task, formal recommendations for official action normally are conveyed to appropriate military agencies by briefing or Disposition Form.



Accession For	
NTIS CRA&I	<input checked="" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	

FOREWORD

In the face of steadily increasing training costs, higher levels of performance must be generated by fewer personnel undergoing training. ARI's research trust of Train the Force supports the development of improved training methods and the application of new instructional technologies that will enable the Army to meet the challenge. Both Smart Technology for Training and Use of Intelligent Computer-Assisted Instruction (ICAI) in Training Systems research areas exploit advances in computer technology and cognitive science techniques to meet the Army need. As development of courseware has been identified as a significant bottleneck in the development of computer-assisted instructional systems, determining the state of the art in computerized aids for authors, as well as providing a forum for the exchange and synthesis of related ideas, was established as an important subgoal of this research. It was met by the ARI sponsorship of the Automated Author Aiding Systems conference held at UCLA on June 6-7, 1984. This document reports the results and conclusions of the conference.



EDGAR M. JOHNSON
Technical Director

Executive Summary

This report documents the proceedings of the Automated Author Aiding Systems Conference held at UCLA on June 6-7, 1984. The purpose of the conference was to determine the current state of the art in computerized aids for authors by providing a forum for the exchange and synthesis of ideas, as well as the demonstration of extant systems. Individuals invited to the conference included researchers from the Department of Defense (DoD) and the academic sector with interests in the area, representatives of DoD agencies and other organizations with interests in the development and application of such systems, and representatives of vendors of author aiding systems.

The conference highlighted issues which must be addressed in the further developments of author aiding systems. These include:

1. The technical adequacy of ISD. While specific concerns were raised about the ISD model, there was general agreement that some type of logical analysis and development procedure such as ISD is necessary to design any instruction. In fact, there is some evidence that applying ISD procedures to courses before they are presented on computers may actually account for their success rather than the computer technology itself (see, for example, the Orlansky and String's review of computer based instruction in the military). While there is evidence to believe that ISD has value as a technical approach, certain limitations and technical difficulties were consistently noted by conference participants. For example, while the procedural steps in ISD are logically organized, some of the blocks within the ISD process require complex and difficult decisions, particularly the initial blocks of front-end analysis activities, involving task analysis and instructional objectives definition. These analysis activities are often not straightforward, particularly for complex cognitive job tasks; further complicating the analytic work is this practical circumstance that access to subject matter experts is usually quite limited. Furthermore, it is often hard for subject matter experts to articulate the key features of a task or to describe standards for determining the adequacy of task performance. This is especially true for cognitive tasks requiring higher order processing of information and decision-making. Greater attention needs to be directed toward articulation of how to perform ISD procedures more effectively.

2. The implementation of ISD. One thread that ran throughout the comments of many participants was the administrative and management problems associated with the implementation of ISD which have impeded its success. These problems include inadequate training of ISD managers and practitioners as well as inadequate resources (including time and expertise) applied to ISD procedures. In addition, communication with job managers in the field, as well as scheduling constraints and availability of subject matter experts contribute further to problems in the effective implementation of the ISD model.

3. Authoring by individuals with team review. While individuals typically write initial drafts (e.g., a task analysis, training objectives, lesson plans, a test), their work is generally reviewed extensively both by other individuals and by committees. The consensus view appeared to be that, not only would intelligent computer aiding contribute to the productivity and effectiveness of individual authors, but that the review process could be made shorter and more efficient with an aiding system that incorporated such features as E mail and word processing.

4. The role of cognitive science. There is a great deal of talk about the importance of incorporating principles of cognitive science into the ISD process or of basing our notions about instructional development on ideas emerging from cognitive science and the study of human information processing. However, it is very difficult for instructional developers and users of ISD to articulate exactly how specific principles from cognitive science were or could be incorporated into their instructional development activities. More attention needs to be devoted to this issue.

5. The range of training problems. Sources of training problems not only include problems stemming from the ISD process itself, but also include problems arising from a broad range of other sources specifically including selection and placement problems (e.g., trainees with inadequate entering skills and knowledge), resource allocation problems (e.g., insufficient time for accomplishing training objectives), and management problems within the employing organization rather than the training organization (e.g., time lag between training and job assignment resulting in skill decay, mismatch between training and job requirements). Effort should be directed to examining the nature of "training" problems beyond the context of ISD narrowly conceived and alternative solution strategies considered.

6. The need for a systems approach. There is a need for ISD to be incorporated into a broader systems view so that ISD does not start with the assumption that training is the appropriate solution to a given problem or, equivalently, so that ISD does not stop with training. A true systems approach would consider the full range of personnel and organizational management, not just the contributions of training and would provide a mechanism for helping to decide if the cost-effective solution is to seek better training. In a given situation it might be, for example, that the cost-effective approach is to seek better personnel selection, or better classification, or better job design, or better equipment design, or better job aids, or better supply schedules, etc. Thus, ISD ought to be viewed in a context broader than training.

7. The capabilities of extant hardware and software. There is a common perception that efforts in training lag behind the computer revolution in software and hardware. However, there are ISD needs that far outstrip the capabilities of current technology. This situation is particularly true in the field of artificial intelligence and has direct implications for the development of an appropriately powerful AI system for ISD.

8. The role of artificial intelligence. While artificial intelligence (AI) is often viewed as having great potential for application to the ISD process, there may be limitations. If ISD is conceived as common sense made systematic, then it may be anomalous to consider AI as a promising approach in ISD since the common lament among AI workers is their inability to capture "common sense". It is also important to consider the costs of adding AI to the ISD process. That is, it will be important to estimate whether the costs incurred are compensated for by increased instructional effectiveness or increased productivity among instructional developers.

In addition, the conference identified foci which should guide the development of automated author aiding tools. These were:

- o a focus on individual worker rather than group training;
- o use of a technical "hard" skill domain such as operator or maintenance training rather than a "soft" skill domain such as leadership;
- o requirement for adequate rather than optimal instruction;
- o in-school instruction rather than unit training;
- o off-the-shelf hardware rather than hardware development;
- o examination of existing systems and identification of needed enhancements;
- o consideration of artificial intelligence techniques particularly for the knowledge base of instructional principles;
- o rapid development with formative evaluation and revision rather than extended development cycle.

CONTENTS

	Page
Executive Summary	v
Introduction	1
Session I: Computer-Based Instruction: Will It Improve Instructional Quality? Commentator: Walter Dick, Florida State University Presentors: William Montague and Wallace Wulfeck, Navy Personnel Research and Development Center	3
Session II: General Issues in the Design of Automated Authoring Aids Chairman and Commentator: M. David Merrill, University of Southern California	5
Automated Instructional Development Aid (AIDA) Mara Marken, Northrup Corporation	5
Effective Learning & Computer-Based Training (CBT) Marcia Watson, Xerox Learning Systems	6
Item Writing & Test Interpretation Gale H. Roid, Western Psychological Services	7
Session III: Automated Aids for Instructional Design Chairman and Commentator: Robert Davis, University of Illinois	8
Application of Software Development Tools to Courseware Authoring Stephen L. Goldberg, Army Research Institute	8
The PLATO Authoring System Edwin P. Durrall, Control Data Corporation	9
Experience with the ZOG Human-Computer Interface System Robert Akscyn, Carnegie-Mellon University	10
The Maintenance Information Authoring System Stephen Morriss, Texas Instruments	12

CONTENTS (Continued)

	Page
Session IV: Issues in Authoring Aids for Document Design	14
Chairman and Commentator: Thomas Duffy, Carnegie-Mellon University	
Computer Aided Authoring of Procedure Training Aids	14
Richard Braby, Naval Training Equipment Center	
WANDAH - A Computerized Writer's Aid	16
Morton Friedman, University of California	
A Software/Documentation Development Environment	
Built from the UNIX Toolkit	16
Robert J. Glushko, AT&T Bell Laboratories	
The UNIX WRITER'S WORKBENCH Software: Computer Aids	
for Text Analysis	17
Stacey A. Keenan, AT&T Bell Laboratories	
Session V: Application of Artificial Intelligence to Automated	
Authoring	18
Chairman and Commentator: Wallace Fuerzeig, Balt, Berarek and Newman	
The Boeing Company Training Equipment and Simulator	19
Story	
J. Marty Jensen, Boeing Aerospace Company	
A Schema for ICAI on Troubleshooting	20
E. Webb Stacy Scientific Systems, Inc.	
Automated Author-Aids for Learning from Cases in	
Ill-Structured Knowledge Domains	21
Rand J. Spiro, University of Illinois	
Economics of Author-Aiding	22
Andrew S. Gibbons, WICAT Systems, Inc.	
Small Group Discussions and Conference Summary	22
Selected References	28
Appendix	A-1

AUTOMATED AUTHOR AIDING SYSTEMS CONFERENCE

Final Report

Introduction

This report documents the proceedings of the Automated Author Aiding Systems Conference held at UCLA on June 6-7, 1984. The purpose of the conference was to determine the current state of the art in computerized aids for authors by providing a forum for the exchange and synthesis of ideas, as well as the demonstration of extant systems. Individuals invited to the conference included researchers from the Department of Defense (DoD) and the academic sector with interests in the area, representatives of DoD agencies and other organizations with interests in the development and application of such systems, and representatives of vendors of author aiding systems.

Approximately 75 individuals attended the conference. (A list of participants is included in the Appendix to this report.) The two-day conference agenda was organized into panels of presentations chaired by a civilian expert, system demonstrations, small group discussion sessions and presentations, and a conference summary. (The conference agenda can also be found in the Appendix.)

The conference was opened by Dr. Harold F. O'Neil, Jr., Director of the Army Research Institute's Laboratory for the Behavioral and Social Sciences. He indicated that the problem stimulating the Army's interest in automated author aiding systems is the lack of efficient methods for creating large amounts of computer-assisted instructional (CAI) materials and the failure of Instructional System Development (ISD) procedures to yield high quality, effective instructional materials. He framed the objective of the conference as a state of the art assessment of online aids

to create online instruction, focusing on the questions of: What do we currently know? and What should we do? Dr. O'Neil identified the eventual products of the conference as an Automated Authoring System for CAI, a research agenda for incorporating artificial intelligence techniques and methods into automated authoring aids, and the revision of ISD policies and procedures. (A copy of Dr. O'Neil's briefing charts are included in the Appendix.)

The following sections of this report summarize the comments of conference presenters. They are organized around the five panels centered on particular aspects or features of automated author aiding systems. Session I, chaired by Dr. Walter Dick of Florida State University served as an overview of the range of issues and examined the potential of computer-based instruction for improving instructional quality. Session II, chaired by Dr. M. David Merrill of the University of Southern California, addressed general issues in the design of automated authoring aids. Session III, chaired by Dr. Robert Davis of the University of Illinois, concerned specific automated aids for instructional design. Session IV, chaired by Dr. Thomas Duffy of Carnegie Mellon University, focused on issues in authoring aids for document design. Session V, chaired by Dr. Wallace Fuerzeig of Bolt, Beranek, and Newman, examined future directions for automated author aiding systems including applications of artificial intelligence. As a conclusion, a summary of the small group discussions and the overall conference is provided. (Hard copies of the viewgraphs used by presenters may be requested from CSE or from the authors directly.)

Session I: Computer-Based Instruction: Will It Improve Instructional Quality? by William Montague and Wallace Wulfeck, Navy Personnel Research and Development Center.

The use of modern computer-based gadgets for instructional purposes is being advocated widely. This advocacy is based mostly on the assumption that computer-based instruction can solve many of the nation's educational problems. In newspapers, magazines, and on television, advertisements suggest that computer-based instruction is effective, fun, can answer concern for the "decline" in educational quality, and provide business, industry, and the military efficient and effective means of training personnel. Although use of this gadgetry will continue to increase simply because of its availability and declining costs, we are dubious about the assumption or hope that its presence will improve the quality of education and training, at least over the near term. It is not that computer-based instructional (CBI) systems cannot be effective; they can be, and some are. Recent reviews thoroughly summarize many demonstrations of the effectiveness gains but, while affordable hardware is a necessary ingredient for widespread effectiveness of CBI, it is not sufficient. Several other ingredients are necessary: good instructional design which uses computer power in appropriate ways, supportable and transportable software, and attention to the ongoing instructional systems into which CBI may be inserted.

We believe that improvement of instruction through CBI will be a relatively slow, evolutionary process. Reasons for this are that: (1) instructional quality is difficult to achieve regardless of the method of delivery, (2) computers, as instructional tools, are in a rudimentary state of development, (3) improvements in either instructional design or computer-based delivery will depend on fundamental changes in the scientific base, and (4) systematic planning for acquiring, standardizing and distributing proven instructional programs, and for incorporating them into schooling, has not been done.

First, we review some attempts to improve quality through the development and implementation of systematic approaches to instructional design. These attempts have not been very successful, and their refinement will take time. Second, we briefly review some developments that have led to common forms of CBI and show that the advantage of using CBI is often unclear. Third, we show that the problems with both instructional design and with traditional CBI are due largely to shortcomings in the underlying scientific base of the psychology of learning and instruction. Fourth, we describe developments in cognitive and instructional psychology and in computer science and artificial intelligence which provide hope that a better scientific base will develop; and we describe particular CBI systems which have been built on these developments. Finally, we discuss some things that can be done to provide a means for the widespread distribution and life-cycle support of CBI systems that have been found to be effective for specific purposes.

What Needs to be Done to Obtain Quality Improvement? We repeatedly make the point that computers and CBI software should not be bought with the expectation that they will solve today's instructional problems. Instead, CBI programs must be built explicitly to teach, and this means

that some form of ISD must be performed. That is why this presentation includes both ISD and CBI as topics; for quality, they can't be separated. Moreover, if the quality of ISD is not improved, then CBI will only improve through sporadic artistry. Therefore, in our view, two developments are necessary: automated aids for ISD and a coherent software distribution and maintenance system.

Automated Aids for Instructional Design and Development. The continued lack of progress with ISD implementation seems due to the variable quality of those doing the implementation and the lack of "how to do it" procedures in usable forms. Since added proceduralization is unlikely to be useful, it is necessary to provide job aids. In paper form job aids can help, but too much still depends on learning by the developers, and there is little time or resources for this. Computer-based aids to authoring instruction can make substantial differences in the quality of instruction whether it is on-line or off-line and, of course, in the efficiency of ISD.

The ISD process has formidable record-keeping problems. A typical military training program has hundreds or sometimes even thousands of learning objectives which must be developed, cross-referenced, tested and taught. Records also cover a wide variety of other ISD activities such as generating test items, choosing alternative training media and strategies, evaluating graduates, and revising courses.

More importantly, computerized ISD aiding systems can not only assist with these record-keeping problems, they can facilitate the development process itself by providing guidance for test and instructional development and similar tasks. Moreover, computer-based systems can ensure that guidance is followed by monitoring and evaluating developers' performance, especially by forcing attention to the delivery options available and the trade-offs among them, and by assisting developers as they proceed. Computer systems can also provide training for instructional developers who can fit it into their work schedules. Finally, these systems are essential for aiding implementation and utilization of CBI. Today, most CBI users such as teachers, school boards, military training activities and businesses want the ability to customize and adapt instructional software. They cannot do this unless tools are available for modification and refinement of CBI programs.

The most desirable reason for using computer tools, however, is that the modern design and development tools can be incorporated into the ISD process. This provides the only mechanism whereby developing science can improve day-to-day practice. Computer-based author aids, then, must be developed to support both ISD for CBI and utilization of CBI.

Improved Distribution and Maintenance of CBI Programs. The transfer of software is a problem. There are attempts to catalog programs and annotate them. While these efforts serve to acquaint people with the range of programs available, several major problems remain. First, many schools, businesses and military agencies are buying computers. Unfortunately, these are often not compatible, so common software packages cannot be developed and shared without substantial recoding and duplication of effort. A second problem is the tremendous repetitive nature of the programming efforts. One programmer or another develops programs that present questions and require selected answers or fill in answers, present text descriptions, etc. All of these require functionally the same code. Computer software companies have recognized the same sort of repetitiveness in computer programs and are developing ways to use codes already developed in new programs. This speeds up the development process and reduces errors

by a large factor. A third problem is that most current CBI programs are not supported by appropriate authoring support and instructional management aids.

We believe that a solution to these problems is to develop families of CBI software to support computer-based instruction in a wide variety of education applications. This can be done by developing libraries of computer-based instructional programs, sufficiently flexible to support development, delivery and management to meet any instructional requirements. The library should also be concerned with demonstration of and specifications for generic hardware systems capable of executing library software, and with planning for and assisting institutionalization of CBI programs. By providing transportable, carefully tested CBI software and development tools, compatibility and supportability problems are solved, user requirements are more efficiently addressed, implementation and life-cycle costs are reduced, standard data on student performance and CBI cost effectiveness can be obtained for budget justification, and acquisition costs of training can be reduced. And again most important, institutional software libraries can achieve a "critical mass" so that evolutionary improvements through application of new technologies like authoring aids can be achieved.

Session II: General Issues in the Design of Automated Authoring Aids

The three papers in this session examined critical issues in the design of automated authoring aids for developing instruction and assessment. Dr. Marken highlighted the time and resource constraints under which training development, particularly by private contractors, typically occurs and their implications for authoring aids. Dr. Watson stressed the importance of considering subject matter content and learner characteristics in designing effective computer-based training. Dr. Roid discussed the use of computerized item writing aids, adaptive and diagnostic testing methods, and automated test interpretation systems to assist in the design and use of assessment.

Automated Instructional Development Aid (AIDA) by Mara Marken, Northrup Corporation

Thousands of training courses are developed annually in the military and industrial communities. Most often these programs are developed by

subject matter experts (pilots, mechanics, accountants, etc.) who have no background in instructional design methodology. This inexperience, coupled with constraints of short development lead times and limited funds and manpower, make training development a difficult and inefficient enterprise.

The automated instructional development aid (AIDA) was designed to meet the needs and skill level of subject matter experts. It maximizes the use of their technical knowledge while minimizing the demand for expertise in instruction design. Simple procedures are easy to learn and the tutorial software enables the user to conduct a task analysis, determine media requirements, and develop lesson plans without a formal training course.

AIDA acknowledges both the constraints of the political/economic environment and the tendency for subject matter experts to base decisions on intuitive feelings and past experience. These liabilities are transformed into assets by encouraging users to express their frustrations and intuition. This fosters creativity while providing documentation for further analysis and informed decision making.

AIDA can be used in conjunction with many of the other automated systems now available although its focus on lesson plan development is unique. It automatically produces performance objectives, course outlines, training standards, course charts, plans of instruction, instructor supplements and media specification.

Effective Learning and Computer-Based Training (CBT) by Marcia Watson,

Xerox Learning Systems

As with any instruction, computer-based training (CBT) needs to take into consideration both learner and subject-matter issues. However, in the case of CBT as an individually delivered form of instruction, the inclusion of both learner and content issues is especially important. Content needs to be carefully analyzed for processes required for learners to acquire and utilize subject-specific data, subject-specific algorithms, and transferable concepts and algorithms. Learner issues that should be considered in designing CBT include emotional set and types of feedback (social, structural). Both learner and content issues need to be considered if effective CBT is to be designed, in that learners with positive emotional sets and appropriate feedback will learn more (especially in independent learning situations) than learners who must attend to their emotional hemostasis and for whom feedback is irrelevant.

Subtly important to the design of CBT are top-down/bottom-up issues in instructional design and learning. Instructional designers will often take the approach that it's simple to learn a given topic because the instructional designer has access to the subject matter as a whole and can see it in an organized pattern. It takes effort on the instructional designer's part to keep the learner's perspective of relative overload and confusion in mind. The designer also needs to remember that learners will seek structure in the discrete experiences involved in their learning. Further, it is wise to help shape learners' views of the topic's structure so as to maximize the probability of functional concepts being formed that will hold water in a variety of situations. Finally, it is important for a designer to maximize learners' focus on content and minimize their attention to hardware/software mechanics.

In short: (1) time "wasted" on learner feelings yields performance

results; (2) knowing learners' natural reward structures helps in designing feedback; (3) it is more productive to aid and abet learners' search for meaning in content than to ignore it; and (4) CBT is most effective when its design considers both learner and content factors.

Item Writing and Test Interpretation by Gale H. Roid, Western Psychological Services.

Three areas of research and development that should be monitored by designers of author-aiding systems are: (1) computerized item-writing aids, (2) adaptive and diagnostic testing methods, and (3) automated test-interpretive systems.

Item Writing. Two trends can be seen in item-writing methods: (a) a movement toward diagnostic testing and test scoring based on the strategies used by examinees in solving items of the problem-solving type, and (b) increased interest in the dimensionality of tests with the advent of factor analysis for dichotomous items. Author aids could incorporate routines for diagnostic scoring (examining patterns of wrong answers), and aids for assessing dimensions underlying item pools. Also, item readability analyses, the "ambiguity" index proposed by Goldberg, and the new domain-referenced test software recently developed by Nitko and associates at Pittsburgh should be investigated, and an automated "test critic" with item analysis developed.

Adaptive Testing. Aids for authors developing adaptive testing based on ISD models have been slow in emerging, perhaps due to the technical nature of the mathematical models underlying these tests. However, with Rasch software and software that can implement estimators of 3-parameter models appearing now for microcomputers, adaptive testing may finally be feasible.

Computerized Test Interpretation. In the field of psychological assessment, a wide variety of techniques and software are developing for the automated interpretation of patterns of test scores, resulting in elaborate narrative reports for individual examinees. To my knowledge, these methods have not been applied to achievement testing or computerized test-like exercises that are embedded in computer assisted instruction. However, the potential is great for producing printed reports for instructors that "talk to them" about the performance of individual trainees. There are at least three levels of interpretive reports: (a) descriptive, which has logical rules such as "this score is above average", (b) "diagnostician modeled" in which the interpretive rules of a skilled instructor/evaluator are automated so that the resulting printed report reads in a way that the instructor would have written it if done individually, and (c) fully "actuarial" in which research-based decision rules (quantitative, not qualitative) are automated. The "actuarial" method, similar to that used for the MMPI and other heavily researched instruments, would make predictive statements such as "this student may have difficulty with the next _____ course, given that he failed the _____-type items in this program-" (where research has shown a correlation between performance on the CAI lessons and future courses).

Session III: Automated Aids for Instructional Design

The five presentations in this session concerned specific systems that have been designed to aid authors in instructional development. Dr. Goldberg described the use of a software engineering tool, Structured Design and Documentation Language (SDDL), to produce general lesson design schemas which can be applied across lessons having similar learning objectives. Dr. Durall discussed the features of the PLATO Author System and its capabilities as an automated authoring aid for instructional designers. Mr. Bauer described a project to develop a computer capability using engineering specifications to develop task analyses for the operation and maintenance of equipment by individual soldiers which can serve as front end analyses for instructional development. Dr. Akscyn described the ZOG system which is a general-purpose, human-computer interface system which has been used as a training system as well as for database systems, management information systems, document management, software management, and electronic communication. Mr. Morriss described the Maintenance Information Authoring System (MIAS), a software development system for creating an accurate and complete data base of procedural information which can be used in instructional delivery and job aiding systems.

Application of Software Development Tools to Courseware Authoring by Stephen L. Goldberg, Army Research Institute

Custom computer-based training courseware that deviates significantly from text-based didactic instruction will be expensive and time consuming to produce given current authoring aids. The Army Research Institute and Scientific Systems, Inc. have the problem of producing a large amount of courseware, in a short period of time with limited resources. We decided to look toward methods being used in software engineering for the means to make the courseware authoring process more efficient and less costly.

In most software development environments, when a series of similar

programs must be written, a general program, called a program schema, is developed. Different versions of the general program are then produced, and great efficiencies can result. In effect the computer is doing most of the detail work.

Good software designers differ significantly in their approach from novices. Experts are much more likely to take an approach that is top-down and hierarchical, rather than solving little pieces of the problem; breadth first, so that all problems at a given level in a hierarchy are solved before going to the next level; modular, such that experts wind up with quasi-independent "packages" instead of highly interdependent subdivision; and reducible to solutions that use existing code or algorithms.

In looking at the instructional design process, we decided that it too lent itself to a top-down, breadth-first modular approach that results in general lesson design schemas which could be applied across lessons having similar learning objectives. A software engineering tool, Structured Design and Documentation Language (SDDL) was used to facilitate the design process and produce documents that communicated lesson flow and parameter requirements to courseware developers. Each module in SDDL represents a section of code that may be developed without worry about how it will fit into the entire lesson. Once a module is coded and running, other modules which interact with them are easier to code and debug.

Within the courseware development project currently underway a number of lesson schemas have been written. Each incorporates a lesson design strategy that, with small variations, can be applied to a number of lesson segments that serve the same function for different systems or pieces of equipment. The use of the lesson schemas allows for quick generation of specific lessons through insertion of appropriate content.

The courseware that results from the process described above has the added advantage that students' interactions with each type of lesson is standardized so that the student can concentrate on the lesson content and not his interaction with the computer. It is currently an open question whether intensive interaction with a training program produced in this way will result in student boredom. For the present program in which reserve soldiers will interact with the system for at most 4 hours a month the question is moot.

The PLATO Authoring System by Edwin P. Durrall, Control Data Corporation

The PLATO Author System and its capabilities as an Automated Authoring Aid are described. The PLATO Author system uses the concept of the Lesson Model in reducing the cost and complexity of computer-based education (CBE) courseware development. A Lesson Model is a software utility that provides certain features including:

- predetermined instructional strategies
- advanced graphics and text editors
- direct-display editing
- preformatted screen displays in basic models
- menu/prompt selection of branching and judging options
- "programmerless" data entry features
- color, animation, and videodisc capabilities in more advanced models

The PLATO Author System currently consists of two sets of authoring utilities: PLATO Courseware Development (PCD1) and PLATO Courseware Development and Delivery (PCD2).

PCD1 is a family of totally programmerless Lesson Models that offers microcomputer-based authoring and delivery; i.e., neither authoring nor delivery requires access to central PLATO. Seven Lesson Models are available with a wide variety of flexibility and ease-of-use features.

PCD2 requires access to central PLATO for authoring, but allows lessons to be downloaded to disks for delivery on a variety of microcomputer systems. PCD2 offers more flexibility than PCD1 in that in addition to certain programmerless features, the full range of capabilities of the PLATO Author Language and the micro PLATO Author Language (formerly known as TUTOR) are available through PCD2.

Further enhancements to the PLATO Author System to offer automated authoring aids in the area of CBE courseware design, as well as development and delivery, are anticipated in the near future.

Experience with the ZOG Human-Computer Interface System by Robert

Akscyn, Carnegie-Mellon University.

ZOG is a general-purpose, human computer interface system based primarily on the concept of menu-selection, with a large database of menus and rapid response to selections. ZOG is intended to be used by both novice and expert users, providing a single interface mechanism that integrates all the computer functions needed by the user.

The basic unit of representation in ZOG is called a frame. Originally, the notion of a frame meant a "structured screenful", i.e., everything the user could see on the terminal screen at one time. Nowadays, with the advent of high-resolution screens, implementations of ZOG provide for several frames to be displayed simultaneously on the screen. A ZOG frame consists of a set of items of different types, each of which carries its own positioning information.

There are three types of interactions with ZOG: navigation, invoking programs, and editing. The default mode of interaction is navigation, in which the user makes a selection via the keyboard or pointing device (mouse), and the system then responds by displaying the next frame. Most selections lead to other frames, but some have "actions", which perform a procedural function such as running a particular program. Finally, the user can enter the frame editor at any frame and make changes to the frame (if he has the requisite privileges, protection being implemented at the level of the individual frame).

The ZOG system is based on a broad set of principles about desirable features of a human-computer interface including:

- total environment
- flexible, efficient tool
- direct manipulation of data
- semi-automatic operation
- low learning overhead
- safe, exploratory environment.

A major component of a ZOG system is its database, which is somewhat different from the traditional notion of a database. The following principles govern the design of the database:

- large size
- shared by multiple users
- generality of representation
- network structures

- tree structures
- menus.

ZOG also makes a major commitment to a particular style of interaction between the user and the system, as indicated by the following features:

- menu selection
- fast response
- browsing
- active selections
- no hidden selections
- common commands
- editor
- no scrolling

A ZOG system needs more than just a database with an interface -- it needs some mechanism for extending the system to provide new functions for the user. The following features describe how this is done:

- mapping data structures
- embedded programs
- environment frames
- frames for input/output.

ZOG has been used in a variety of applications, including instruction/training. ZOG has been used as a training system in several capacities: (1) providing on-line help; (2) as a guidance system for using other on-line systems by shielding users from the idiosyncracies of a particular interface; and (3) as an index and control mechanism for a videodisc player.

We believe that ZOG has many strong points; in fact, it has often surprised us (even with our understandable optimism) how well ZOG has adapted to the many demands we have made of it. The most important of the strong points are described below:

- robust enough to be put into operational use
- easy for computer novices to learn and use
- users do not outgrow ZOG as they become expert
- can assimilate and integrate many different applications
- supports database browsing
- supports large databases
- can exploit schemas for building databases
- can be used as the sole interface (shell) for an operating system
- has a simplified model of window use
- has a simplified model of multiple processes
- can be used as an interface mechanism for video discs
- can make good use of a pointing device (mouse)
- can exploit distributed systems
- supports a community of communicating users.

Below are the weaknesses of ZOG that we have become aware of:

- ZOG sacrifices efficiency of particular applications to get integration
- ZOG does not support a fast database query language
- inexperienced users can get lost
- biased too much toward breadth-first view
- ZOG depends too critically on the speed of the disk technology
- can't represent all states of a complex, unordered task environment
- can't handle highly dynamic data
- lack of graphics and multiple fonts
- ZOG cannot be used over standard telecommunication lines.

There is now work going on outside CMU to create a follow-on to ZOG called KMS (Knowledge Management System). KMS is designed to remedy many of the shortcomings of the ZOG implementations, and specifically to make full use of the potential of high-resolution display technology. Some of the features of KMS that represent improvements over ZOG are the following:

- Graphics (lines, rectangles, curves, picture images) and multiple fonts.
- Greater use of the pointing device to specify objects and parts of objects to operate on, and screen cursor images to provide feedback of the current system context.
- Direct output of good quality hardcopy (no need for a separate formatting system that operates as a post-processor).
- Copying material easily across frame boundaries.
- Additional intrinsic views of the database -- specifically, a depth-first view.
- Closer integration of the editor with the rest of system, to make editing seem more natural, rather than a special mode that one is continually entering and leaving.

The Maintenance Information Authoring System by Stephen Morriss, Texas Instruments.

The Maintenance Information Authoring System (MIAS) is a software development system dedicated to the construction of accurate and complete procedural information, for use by noncomputer knowledgeable technical writers, that conforms to the special requirements and constraints imposed by an equally dedicated delivery medium. This paper describes, beginning with the requirements and constraints of the delivery medium, the process and structure of authoring system specific maintenance instructions.

In 1979, the Department of Defense, sponsored the development of a new way to deliver information to military technicians. The intent of this new system was to provide individuals with a portable tool with which the information/knowledge requirements to operate and maintain the new weapon systems could be obtained at the time and place the individual needed it. The base level requirements for this new information delivery medium were that it be man portable, rugged, easy to use, and easy to update the information delivered.

Our response to this challenge was the Personal Electronic Aid for Maintenance (PEAM). PEAM is a suitcase size computer, weighing near 25 pounds. The suitcase contains a detachable man/machine interface that contains a 5 by 7 inch flat panel electroluminescent display. The display is capable of matrix graphics on a 352 by 512 grid. The line density is 71 lines per inch. The system has the capacity to read text aloud by means of text-to-speech synthesis and to respond to voice command by the system operator. Information storage is by means of 1 megabyte solid state CMOS static RAM cartridges. The system is capable of battery operation for up to 4 hours and is compatible to standard 110 current. The system is ruggedized to withstand hostile environments including moisture and temperature extremes.

The information storage, being limited by the size, environmental, and power constraints, required the development of a highly specialized database design that would allow for the compression of the equivalent of 3-4000 pages of text and graphics within the megabyte cartridge.

At the same time we were developing the PEAM system, we began the design of a method to develop the information to be delivered. Because PEAM was considered an instructional delivery system, our starting point was the existing CAI authoring systems currently in use.

As we progressed in our design, it quickly became apparent that there existed a marked difference in the methods used in CAI authoring and the type of information that we intended for PEAM. Specifically, CAI systems are, by design, generic tools for mapping information into one or many learning strategies. The tools must be general purpose, because the domains of knowledge or the subject matter vary in their structure, their completeness, their vocabularies and their level of abstraction.

PEAM information, on the other hand, is composed of only one kind of learning strategy, step by step instruction. The information structure and the degree of completeness is fixed by the structure of the system to be maintained. The vocabulary, we discovered, is relatively constrained, containing approximately 1300 generic terms (exclusive of part names). The level of detail had to be at a performance or action level. Graphic data has two functions: 1) to provide for location discrimination; and 2) to provide for identification of specific parts.

We designed an authoring system that is capable of performing to the following requirements:

1. Takes full advantage of the inherent structure of the maintenance domain by:
 - A. Providing assistance in defining the system and information structure.
 - B. Providing assistance in the control of system and generic vocabulary, syntax, and thus, the level of detail for each instruction.
 - C. Providing assistance in development of the troubleshooting information.
2. Takes full advantage of available source documentation by:
 - A. Maintaining source document and graphic data control.
 - B. Allowing for direct information capture.
 - C. Allowing the easy trace of information change based upon changes in the engineering of the system.
3. Takes full advantage of the structure for authoring by:
 - A. Providing for assistance in setting up the job.
 - B. Providing assistance in management of the job.
 - C. Provides for the most skill at the highest and less burdened activities of the job and the least skills at the lowest and more numerous tasks of the job.

The objectives for MIAS include:

- To provide a usable system for primarily defense contractors.
- To provide for low cost acquisition and operation.
- To provide for a flexible user base from experienced to inexperienced writers.
- To provide for consistent quality products.
- To be easy to learn and use.
- To provide management of the ability to observe the total picture of the job process.
- To provide for product cost control.

Session IV: Issues in Authoring Aids for Document Design

The papers in this session examined issues in aiding systems for authors of documents. These issues and systems are relevant to the problem of instructional design since a great deal of instruction both offline and online is text-based. Mr. Braby described computer aided authoring routines to create Procedure Training Aids which present textual material in an instructional format and direct students how to use the material. Dr. Friedman presented WANDAH (Writing Aid AND Author's Helper), an intelligent instructional system designed to assist writers in all phases of writing including planning and organizing ideas, transcribing ideas into print, and editing and revising. Dr. Glushko described a system called SOLID (System for On-Line Information Development) that treats document development and management like software development and allows for development and editing of multiple-author, multiple-version documents with a wide range of edit and control functions. Ms. Keenan described the WRITER'S WORKBENCH programs designed to aid in the editing and revision states of technical documentation.

Computer Aided Authoring of Procedure Training Aids by Richard Braby,
Training Analysis and Evaluation Group, Naval Training Equipment Center.

Learning to operate complex equipment is a common type of military training activity. Training devices and operational equipment designated for training are generally provided to students for "hands-on" practice. This "hands-on" training is expensive and access to the equipment is necessarily limited. To make the best use of this time, students need preliminary training. For example, before first using a Cockpit procedure Trainer, the student Navy pilot studies the NATOPS manual and other printed materials describing a specific procedure. However, we have found that this kind of preliminary training is lacking in a number of important respects:

- o Printed materials (e.g., workbooks describing procedures) are mainly narrative--the operating instructions they teach are heavily visual in nature.

- o Class instruction is usually in a group. However, learning to

perform procedures is frequently more effective when practice is self-paced.

- o Students are frequently unprepared to take best advantage of their limited time with the training equipment. When properly prepared, available "hands-on" time produces greater training gains.

This paper describes a Procedure Training Aid designed to overcome these limitations. It presents the instructional formats for the material, describes how students use the material before going to the trainers, and presents the results of a field trial. Because these materials are fairly expensive to produce, computer-aided authoring routines have been created to reduce cost. These routines are described along with sample materials generated. Finally, recommendations are provided on how best to use computer-generated Procedure Training Aids.

By incorporating learning principles, our Procedure Training Aids direct the student in efficient practice in learning to perform complex procedures. Typically Procedure Training Aids have four types of pages:

- o Presentation Page Presents the acts necessary to perform a step in the procedure.
- o Paraphrase Page Exercises for the student in recalling the content of the previously presented Presentation Page.
- o Road Map Page With guides and prompts, provides the student with an exercise in which he talks and points his way through a group of steps and learns to do this as a smooth sequence of actions.
- o Paper Mock-up Provides the student with an extension of the Road Map exercise, this time without guides and prompts.

For all these pages, visual job task information is presented graphically. Words are used only for those messages that must be transmitted with language. Field tests have shown that students using Procedure Training Aids perform significantly better than students using more traditional types of materials.

Procedure Training Aids are relatively expensive to produce using traditional techniques. Their complex, heavily illustrated pages require considerable time and expense to prepare. In addition, a larger number of pages are required to support a procedure than with conventional materials.

Computer-aided authoring techniques can make the production of these materials economical. Two generations of computer routines have been developed for this purpose, and a third is nearing completion.

These routines make it easy and fast for an author to enter procedure information into a computer data base. Only plain English commands are needed. The computer then automatically formats this information and prints camera-ready copy containing the formatted text and boxes for the graphics which must then be set in place by hand.

The first generation routine is called PLA for Computer Automated Page Layout for Text-Graphic Materials (Terrell, 1983). It is written in BASIC and runs on the WANG 2200 MVP minicomputer. The basic steps in using this routing are:

- o The author enters procedure data which includes:
 - steps in the procedure
 - dimensions of pictures for each step
 - text for labels
 - picture-label relationships
 - page headers and footers.

- o The computer:
 - selects optimum page layouts
 - draws boxes for pictures and text; lines for darts
 - puts text in label boxes
 - types headers and footers.
- o The author then completes the process:
 - resolves layout problems that cannot be solved by the PLA routine
 - places pictures in designated boxes
 - places adhesive-backed darts connecting text boxes with their points of action on the graphics
 - ensures that the material is camera ready.

A second generation routing called MicroPLA is similar to PLA except that it runs on a microcomputer, the Commodore 8032 and contains a large number of default values that reduce the time required to enter data into the system.

WANDAH - A Computerized Writer's Aid by Morton Friedman, University of California, Los Angeles.

WANDAH (Writing Aid AND Author's Helper) is a new generation intelligent instructional system designed to assist writers in all phases of writing -- planning and organizing ideas, transcribing ideas into print, and editing and revising. The design of WANDAH draws on three areas of research: first, the analysis of writing as a cognitive problem solving activity; second, modern composition theory and research; and third, research on the human-computer interface.

WANDAH has three major components:

1. A word processor designed expressly for on-line composing. The word processor includes such features as split screen, extensive on-line keys, easy-to-follow menus, and special, labelled function keys.
2. A set of prewriting aids incorporating principles of modern composition theory that help writers generate ideas and plan their work.
3. A set of aids helping writers review and revise their work grammatically, stylistically, and thematically.

These aids include spelling, word usage and punctuation checkers, a style analyzer, and a commenting facility, as well as aids that create an ex-post facto outline of the work and that examine text for coherence by looking at transition phrases and pronouns.

WANDAH is written in UCSD Pascal and runs on the IBM-PC and similar machines.

A Software/Documentation Development Environment Built from the UNIX*

Toolkit by Robert J. Glushko, AT&T Bell Laboratories

Mismatches in technology, media, and organizations in the development

* UNIX is a trademark of Bell Laboratories. My comments refer to the version known as System V distributed by Western Electric.

of software and software documentation often leave the documentation late, incomplete, and poorly coordinated with the software. One solution to this problem is to treat software documentation like software. SOLID, the System for On-Line Information Development, builds on the popular UNIX operating system and exploits the idea that all information that can be stored on a computer shares a common life cycle and can be created, managed, and delivered with the same methods and commands. SOLID, together with the UNIX system, solves ten key problems of text development and management that can be faced by any project.

SOLID is a "integrated" programming and documentation environment built from the UNIX system's tools. It implements the concepts of Source, where all the source for programs and documents are kept, Product, where all the executable programs and formatted documents reside, and Generation Procedures, which translate all types of Source into all forms of Product. These concepts are implemented as UNIX command language procedures that combine existing tools. SOLID also provides Skeletons, the most common outline form of each type of Source. Developers writing a new module of code or documentation do not start with an empty file. Instead, they are given an outline which reduces drudgery and painlessly enforces some standards of style and content.

SOLID is notable for what it does and what it doesn't do. On the "does" side, it provides a single environment that is being used to develop all documentation for computer systems, all programs, and mixtures of both. It comes with a default environment that is occasionally used unmodified.

On the other side, it is not a monolith that restricts the user to those options that are built in. Since all SOLID source is under SOLID control, the user can change or extend any of SOLID's capabilities. Some portions, in particular Generation Procedures, are intended to be extended to support new Source and Product types. It provides no editor; people use any that they like. It doesn't enforce any particular programming or documentation methodology, although Skeletons have proved useful in encouraging the methodologies and styles a group prefers.

Although released only recently for internal use, SOLID is now used by a dozen organizations at Bell Laboratories. Some use it for their documentation only, others for their entire product. Because it is based on observing the ways people have customized UNIX to suit their needs, it seems to match the way many groups do business. Because it is designed to be modifiable and is supported by its own mechanisms, people easily extend and customize it to suit their own needs. Because it stands firmly on standard UNIX, as UNIX becomes available on new or different hardware, SOLID is also. Finally, it preserves the tool orientation of UNIX. New capabilities added to UNIX are immediately available to SOLID users.

The UNIX WRITER'S WORKBENCH: Software: Computer Aids for Text Analysis by
Stacey A. Keenan, AT&T Bell Laboratories.

The UNIX* WRITER'S WORKBENCH** Software, was developed at AT&T Bell Laboratories to aid the editing and revision stages of technical

* UNIX is a trademark of AT&T Bell Laboratories.

** WRITER'S WORKBENCH is a trademark of AT&T Technologies.

documentation. In designing the WRITER'S WORKBENCH system, we followed six design principles which can apply to many kinds of software systems. We felt the programs should be:

- rational--programs are based on expert knowledge about what makes documents difficult to understand.
- diverse--programs reflect the complexity of communication by reporting only aspects of document style. One score does not describe the quality of a document.
- evaluative relative to standards--documents are judged according to their intended audience.
- modifiable--users can select standards, build new standards, modify dictionaries, and request short or long output.
- specific--programs tell what problems a text has, and shows where those problems occur.
- informative--programs provide information, but do not make rigid decisions, or make changes to text.

The WRITER'S WORKBENCH programs analyze prose documents stored in the computer and suggest improvements to the writing. The user evaluates the suggestions and decides what to change. The programs can be divided into four major areas: 1) proofreading, 2) stylistic analysis, 3) on-line information about English usage and about the WRITER'S WORKBENCH programs, and 4) utility programs, for modifying other programs. The programs are designed to provide as much or as little information as the user requires.

The WRITER'S WORKBENCH system has been widely used at AT&T Bell Laboratories for over 4 years. Freshman composition students at Colorado State University and their teachers are enthusiastic about using the programs. Use of the WRITER'S WORKBENCH system speeds editing, heightens authors' awareness of writing style, and frees the author (or teacher) to concentrate on more substantive issues of document content.

Session V: Application of Artificial Intelligence to Automated Authoring

The papers in this session explored future directions of author aiding systems with a particular emphasis on the potential contribution of artificial intelligence to these systems. Dr. Jensen described research and development efforts to achieve an expert training system technology that is approaching applied artificial intelligence methodology in training systems. Dr. Stacy presented a schema for intelligent computer-assisted instruction on troubleshooting. Dr. Spiro drew the distinction between well- and ill-structured domains of knowledge representation and

application and described an instructional approach, Experience-Consolidation Systems, that relies upon the human visual-perceptual processing system to acquire knowledge in ill-structured domains. Dr. Gibbons concluded the session by documenting the major costs residing in the development and delivery of computer-based training, demonstrating the economic effects of efficient systems, and enumerating the features that future authoring systems must have to be efficient.

The Boeing Company Training Equipment and Simulator Story by J. Marty Jensen, Boeing Aerospace Company.

Over the past 6 years, the Boeing Training Systems Organization has expended research and development effort in achieving an expert training system technology that is approaching applied artificial intelligence methodology in training systems.

Our philosophy is based on the principle of "The End Depends on the Beginning" for the design of our Database Management System Controller Software Executive. The computer driven training database executive is designed with intelligent friendly logic that helps student learning and aiding to task completion through the delivery of the database elements. The database contains the elements necessary to transfer knowledge to users and to guide practice of operating procedures. The database is input by the user and establishes the media for the system executive. A primary requirement for the database involves detailed planning and front end analysis to establish the intelligence for the Expert System. The logic for the Expert System Executive provides instructions help/aiding and remediation to the user during the execution of the learning process. The executive also monitors and maintains records of individual performance.

The Expert System philosophy has been applied in several training system applications. User friendly software is provided for easy input and control of the database and for steering the Expert System Executive for delivery of the database.

Evidence of the effectiveness of training developed using this system emerged from an evaluation of one application - the Maintenance Institutional Trainer (MIT). Two formal classes trained on the MIT. Its training effectiveness was far in excess of the users original expectations. The U.S. Army training community estimated that an additional 4 to 6 weeks of training time would have to be added to the course of instruction if the MIT were not available. An important added benefit was the reduced need to train on tactical hardware. With the MIT available, a class size of 15 students only required the use of two tactical systems while without the MIT a minimum of six tactical systems

would have been needed.

The effectiveness of the trainer provided other benefits also. The CAI feature promoted self-paced training capability. The management feature provided course administrative support. Both features resulted in reduced requirements for instructor and overhead personnel support.

In summary, the Maintenance Institutional Trainer was an extremely effective, state of the art training device that should be further developed for training on additional weapons systems.

During the past year, Boeing has assembled a laboratory trainer generic development station based on lessons learned from previous trainer designs for the purpose of experimental application of technologies in hardware, software, courseware authoring, and training records. The station is used to test and demonstrate various trainer interfaces for interactive student participation. It incorporates 2D and 3D equipment for testing methods and principles for delivery of training sequences. This allows front end application of courseware techniques prior to committing to a specific trainer baseline configuration. A VAX 11/750 is the host processor which contains a trainer executive control system programmed in FORTRAN. It interfaces with a smart I/O system that monitors switch inputs and controls outputs to 3D simulated test equipment displays, analog and digital devices. Standard terminals, a graphic display, a record and playback video disc and various interactive input devices are also included for 2D simulation. A courseware authoring language called ALPHA is the user interface to integrate these devices for delivery of training scenarios. The authoring system is a computer aided process which allows courses to be entered into the computer database for both the 3D and 2D devices. This system is now able to demonstrate and available to evaluate user requirements and future research application.

A Schema for ICAI on Troubleshooting by E. Webb Stacy, Jr., Scientific Systems, Inc.

The purpose of this talk was to sketch features of a system that can provide intelligent computer-aided instruction on troubleshooting. In keeping with the theme of automated authoring aids, the system will ease the process of developing training for troubleshooting a system to the extent that generic troubleshooting knowledge applies to that system.

We envision the system to have three major components: a knowledge base, a model of the student, and an instructional engine. The knowledge base contains the information to be taught, the student model represents the student's status with respect to the knowledge base, and the instructional engine provides instruction with the goal of moving the student model (and therefore the student) in the direction of the knowledge base (and therefore the expert.)

It is necessary that the knowledge representation strategy in the knowledge base have certain features isomorphic to the way the expert actually represents knowledge. An attempt to train medical residents using the rules in MYCIN, a medical diagnostic expert system, showed that the rules were not particularly comprehensible to humans; a reformulation of the rules in a form more natural to humans (NEOMYCIN) resulted in no decrement in performance as an expert system but much more effective training. Hence it is important to understand the nature of expert knowledge.

Expert troubleshooters can minimize total troubleshooting time by considering simultaneously the time costs of tests, the informativeness of tests, and the relative reliability of system components. They have pragmatic, but not necessarily detailed, knowledge of system and subsystem functions, usually have multiple representations of the systems (e.g., logical and physical), and have a large vocabulary of recognizable symptom patterns. This knowledge will therefore be a necessary component of the knowledge base.

The student's knowledge may differ from the expert's in that the student may have missing knowledge. For instance, student troubleshooters have been shown to omit one, two, or all of the time the cost of tests, the informativeness of tests, or component reliability. The student may also have extra (inaccurate) knowledge, which may take the form of overgeneralization, overdifferentiation, or misconceptions. Finally, the student's knowledge is slower and more effortful than the expert's.

The instructional engine will deal with missing knowledge via conventional representation methods; with extra knowledge via debugging of student knowledge using well-chosen counterexamples; and with slow performance via practice with feedback in a simulated environment.

Clearly, there will be features of an ICAI system for troubleshooting a specific system that are unique to that system. We do not believe that troubleshooting knowledge applies mutatis mutandis for all systems. We do believe that there is much commonality in the way expert troubleshooters think about their systems, and in effective strategies for training troubleshooting skills. By arranging an ICAI system as described here, with careful attention to facilitating modification of its system-specific aspects, development effort for training the troubleshooting of a new system will be greatly reduced.

Automated Author-Aids for Learning from Cases in Ill-Structured Knowledge

Domains by Rand J. Spiro, University of Illinois at Urbana.

The distinction was drawn between well- and ill-structured domains of knowledge representation and application. In ill-structured domains there is no suitable set of defining characteristics for category membership, proper hierarchy assumptions are violated across individual cases, and category prototypes tend to be frequently misleading -- in short, general principles are highly misleading by themselves. It was argued that most domains are more ill-structured than is typically acknowledged. In such situations learning is experience-intensive, requiring exposure to large numbers of cases (examples, events, instances, situations, etc.), with rich and highly interconnected memory representation of the patterns of attribute-configurations from the individual cases.

These requirements pose instructional problems because of the demands they place on exposure time, memory load, and sorting out the inefficacious natural arrangement of cases. An approach that overcomes each of these problems was presented: Experience-Consolidation Systems (ECSs). ECSs present in a mnemonically convenient form much of the information that normally is acquired from field experience, except in hours instead of years. They rely on the human visual-perceptual processing system's innate ability to simultaneously represent complex and irregular interrelationships, something for which the more verbal and analytic processing system is poorly suited. Integrated visual analogues of a nearly exhaustive set of perspectives for describing an individual case are

presented and re-presented in varying contexts by the computer using a unique arrangement algorithm that "criss-crosses" the set of cases in such a way as to facilitate the noticing of configural attributes of resemblance within the outcome "families" that need to be distinguished. Thus the performance process engendered is one of perceptual recognition. The principle that enables correct recognition is very similar to that of real gene-pool inheritance and family resemblance.

Among many other uses, ECSs have great potential value in military settings given the poor literacy skills of many recruits and the prevalence of experience-intensive, ill-structured military situations that make the development of systems to consolidate the acquisition of case experience imperative if personnel are to efficiently learn how to make accurate judgments and decisions when rules presented in formal instructional settings do not generalize sufficiently.

Economics of Author-Aiding by Andrew S. Gibbons, WICAT Systems, Inc.

The form and extent of future author-aiding systems will depend on how well they make authoring less costly or more productive.

As the military and business communities have increased their commitment to computer-based training (CBT) by authoring thousands of hours of instruction, it is becoming clear that the major costs of CBT reside in development and delivery and not in hardware. This report demonstrates the economic effect of efficient systems and enumerates features that efficient authoring and author-aiding systems must have.

Small Group Discussions and Conference Summary

Sessions VI, VII, and VIII were devoted to panel discussions of issues emerging from the presentations. Participants were assigned to panels to insure a mix of individuals representing the military, private, and academic sector. Each of the civilian consultants served as the facilitator for one of the panels. The questions which structured these discussions and presentations are included in the Appendix.

Dr. Dexter Fletcher of the University of Oregon served as the conference summarizer. His comments were based on the presentations and panel discussions that were held over the two days. His approach was to identify anomalies that he had noted during the proceedings, as a way of highlighting the issues which must be addressed in the further development and use of author aiding systems. These anomalies include:

1. The technical adequacy of ISD. While specific concerns were raised about the ISD model, there was general agreement that some type of logical, structured analysis and development procedure such as ISD is necessary to design any instruction. In fact, there is some evidence that applying ISD procedures to courses before they are presented on computers may actually account for their success rather than the computer technology itself (see, for example, Orlansky and String's review of computer based instruction in the military). While there is evidence to believe that ISD has value as a technical approach, certain limitations and technical difficulties were consistently noted by conference participants. For example, while the procedural steps in ISD are logically organized, some of the blocks within the ISD process require complex and difficult decisions, particularly the initial blocks of front-end analysis activities, involving task analysis and instructional objectives definition. These analysis activities are often not straightforward, particularly for complex cognitive job-tasks; further complicating the analytic work is this practical circumstance that access to subject matter experts is usually quite limited. Furthermore, it is often hard for subject matter experts to articulate the key features of a task or to describe standards for determining the adequacy of task performance. This is especially true for cognitive tasks requiring higher order processing of information and decision making. Greater attention needs to be directed toward articulation of how to perform ISD procedures more effectively.

2. The implementation of ISD. One thread that ran throughout the comments of many participants was the administrative and management problems associated with the implementation of ISD which have impeded its

success. These problems include inadequate training of ISD managers and practitioners as well as inadequate resources (including time and expertise) applied to ISD procedures. In addition, communication with job managers in the field, as well as scheduling constraints and availability of subject matter experts contribute further to problems in the effective implementation of the ISD model.

3. Authoring by individuals with team review. While individuals typically write initial drafts (e.g., a task analysis, training objectives, lesson plans, a test), their work is generally reviewed extensively both by other individuals and by committees. The consensus view appeared to be that, not only would intelligent computer aiding contribute to the productivity and effectiveness of individual authors, but that the review process could be made shorter and more efficient with an aiding system that incorporated such features as E mail and word processing.

4. The role of cognitive science. There is a great deal of talk about the importance of incorporating principles of cognitive science into the ISD process or of basing our notions about instructional development on ideas emerging from cognitive science and the study of human information processing. However, it is very difficult for instructional developers and users of ISD to articulate exactly how specific principles from cognitive science were or could be incorporated into their instructional development activities. More attention needs to be devoted to this issue.

5. The range of training problems. Sources of training problems not only include problems stemming from the ISD process itself, but also include problems arising from a broad range of other sources specifically including selection and placement problems (e.g., trainees with inadequate

entering skills and knowledge), resource allocation problems (e.g., insufficient time for accomplishing training objectives), and management problems within the employing organization rather than the training organization (e.g., time lag between training and job assignment resulting in skill decay, mismatch between training and job requirements). Effort should be directed to examining the nature of "training" problems beyond the of ISD narrowly conceived and alternative solution strategies considered.

6. The need for a systems approach. There is a need for ISD to be incorporated into a broader systems view so that ISD does not start with the assumption that training is the appropriate solution to a given problem or, equivalently, so that ISD does not stop with training delivery. A true systems approach would consider the full range of personnel and organizational management, not just the contributions of training and would provide a mechanism for helping to decide if the cost-effective solution is to seek better training. In a given situation it might be, for example, that the cost-effective approach is to seek better personnel selection, or better job classification, or better job design, or better equipment design, or better job aids, or better supply schedules, etc. Thus, ISD ought to be viewed in a context broader than training.

7. The capabilities of extant hardware and software. There is a common perception that efforts in training lag behind the computer revolution in software and hardware. However, there are ISD needs that far outstrip the capabilities of current technology. This situation is particularly true in the field of artificial intelligence and has direct implications for the development of an appropriately powerful AI system for ISD.

8. The role of artificial intelligence. While artificial intelligence (AI) is often viewed as having great potential for application to the ISD process, there may be limitations. If ISD is conceived as common sense made systematic, then it may be anomalous to consider AI as a promising approach in ISD since the common lament among AI workers is their inability to capture "common sense". It is also important to consider the costs of adding AI to the ISD process. That is, it will be important to estimate whether the costs to be incurred are compensated for by increased instructional effectiveness or increased productivity among instructional developers.

Future Research

What future directions should research and development efforts in the area of automated author aiding systems take? A view frequently expressed by conference participants was that, on the one hand, it was too early to plan for a fully automated intelligent authoring system. However on the other hand, many computer-based procedures could be put to good use. An Instructional Designer's/Instructor's Workbench serves as a good example. Such a system would eventually include an expert system for instructional design principles, as well as systems for word processing, guidance in instructional design, and template-based author aiding.

To facilitate rapid application of automated author aiding tools, a number of parameters were judged as appropriate constraints for initial development activity. These were:

- o a focus on individual worker rather than group training;

- o use of a technical "hard" skill domain such as operator or maintenance training rather than a "soft" skill domain such as leadership;
- o requirement for adequate rather than optimal instruction;
- o in-school instruction rather than unit training;
- o off-the-shelf hardware rather than hardware development;
- o examination of existing systems and identification of needed enhancements;
- o consideration of artificial intelligence techniques particularly for the knowledge base of instructional principles;
- o rapid development with formative evaluation and revision rather than an extended development cycle.

SELECTED REFERENCES
Automated Author Aiding Systems

- Akscyn, R.M., & McCracken, D.L. ZOG and the USS CARL VINSON: Lessons in system development. Computer Science Department, Carnegie-Melon University, Pittsburgh, Pennsylvania, 1984.
- Andrews, D.H., & Goodson, L.A. A comparative analysis of instructional design. Journal of Instructional Development, 3(4), 2-16, 1980.
- Bass, R.K., & Dills, C.R. (Eds.). Instructional development, the state of the art, II. Dubuque, Iowa: Kendall/Hunt, 1984.
- Bianchi, M.H., Glushko, R.J., & Mashey, J.R. A software/documentation development environment built from the UNIX toolkit. In H.J. Schneider & A.I. Wassermann (Eds.), Automated tools for information systems design. Amsterdam: North Holland, pp. 107-108, 1982.
- Black, J.B., & Sebrechts, M.M. Facilitating human-computer interaction. Applied Psycholinguistics, 2, 149-177, 1981.
- Bork, A., Franklin, S.D., Trowbridge, D.E., Von Blum, R., Katz, M., Kurtz, B.L. Graphics and screen design for interactive learning. Educational Technology Center, University of California.
- Bork, A. Learning with computers. Bedford, MA: Digital Equipment Corp., 1981.
- Bower, G.H. Organizational factors in memory. Cognitive Psychology, 1, 18-46, 1970.
- Brown, J.S., & Sleeman, D. Intelligent tutoring systems. London: Academic Press, 1982.
- Bunderson, C.V., Gibbons, A.S., Olsen, J.B. & Kearsley, G.P. Work models: Beyond instructional objectives. Instructional Science, 10, 205-215, 1981.
- Card, S.K., Moran, T.P., & Newell, A. Computer text editing: An information-processing analysis of a routine cognitive skill. Cognitive Psychology, 12, 32-74, 1980.
- Carter, L.F. The systems approach to education: Mystique and reality. Introduction to the systems approach. (Educational Technology Review Series No. 3) Englewood Cliffs, N.J: Educational Technology Publications, 1973.
- Cherry, L.L. Writing tools: The STYLS and diction programs. Computing Science Technical Report No. 91. Bell Laboratories. 1981.
- Coke, E.U. Computer aids for writing text. In David H. Jonassen (ed.), The Technology of Text: Principles for Structuring, Designing, and Displaying Text. New Jersey: Educational Technology Publications, 383-400, 1982.

- Coke, E.U., & Rothkopf, E.A. Note on a simple algorithm for a computer-produced reading ease score. Journal of Applied Psychology, 54, 208-210, 1970.
- Duffy, T.M., & Kabance, P. Testing a readable writing approach to text revision. Journal of Educational Psychology, 74(5), 733-748.
- Fletcher, J.D. Intelligent instructional systems in training. In S.A. Andriole (Ed.). Applications in artificial intelligence. Petrocelli Books, Inc. 1984.
- Dallman, B.E., Pieper, W.J., & Richardson, J.J. A graphics simulation system--Task emulation not equipment modeling. Journal of Computer-Based Instruction, 10(3&4), 70-72, 1983.
- Dallman, B. Graphics simulation in maintenance training--Training effectiveness at cost savings. In Computer Literacy, Intelligent CAI, 1982 Conference Proceedings of the Association for the Development of Computer-Based Instructional Systems (ADCIS), Bellingham, WA, 1982.
- Frase, L.T., MacDonald, N.H., & Gingrich, P.S. Unix Writer's Workbench Software. Bell System Technical Journal, 1863-1921, 1983.
- Glushko, R.J. & Bianchi, M.H. On-Line Documentation: Mechanizing Development, Delivery, and Use. Bell System Technical Journal, 61, 6, 1313-1323, 1982.
- Glushko, R.J. Text development and management in UNIX-based projects. IEEE Transactions on Communication. 1984.
- Goguen, J., Linde, C. Optimal Structures for Multimedia Instruction, SRI International, 1984.
- Goheen, S., & Jordan, D. Evaluation of TICS: A Multics Sybsystem for the Development and Use of CAI Courseware. MITRE Technical Report 2749, 1974.
- Hendrix, G.G., & Sacerdoti, E.D. Natural-Language Processing Part One: The Field in Perspective. Technical Note 237, SRI International, 1981.
- Hendrix, G.G. Klaus: A System for Managing Information and Computational Resources. Technical Note 230, SRI International, 1980.
- Kaplow, R. A description of basic author aids in an organized system for computer assisted instruction. ARI Technical Report #TR-78-A8. Massachusetts Institute of Technology, 1978.
- Kaplow, R., et al. Teacher-interactive computer system: I. The author-language and instruction manual. II. Language Specifications. Massachusetts Institute of Technology. 1971; (updated 1973, 1974, 1975).

- Kaplow, R., Schneider, D., Smith, F.C., Jr., & Strenstud, W.R. Computer Assistance for writing interactive programs: TICS. Proceedings, Association for Computing Machinery, 1973.
- Kaplow, R., Desch, S.H., Jr., Pettijohn, D.O., Rodman, M.H., & Smith, F.C., Jr. Illustrations of Conversational, Inquiry, Problem-Solvings, and Questionnaire Type Interactions within the TICS System. Proceedings, Seventh Annual Princeton Conference on Information Sciences and Systems. 1973.
- Kearsley, G., Hunter, B., & Seidel, R.J. Two decades of computer-based instruction projects: What have we learned? T.H.E. Journal, 1983.
- Kottenstette, J.P. CBI Applications: An organization perspective Denver Research Institute, University of Denver, 1984.
- Lamos, J.P. The development of an instructional design editor, Denver Research Institute, University of Denver, 1984.
- Lamos, J.P. Progress Report: The HELP Authoring System. Denver Research Institute, University of Denver, 1984.
- Lamport, L., Goguen, J. Phil: A Context-Sensitive Structured Graphical Editor. SRI International, 1981.
- Logan, R.S. Instructional systems development: An international view of theory and practice. New York: Academic Press, 1982.
- MacDonald, N.H. The UNIX Writer's Workbench Software: Rationale and Design. Bell System Technical Journal, 62, 6, pp. 1891-1908, 1983.
- Malfero, A.J., McNabb, S.D., Joyce, R.P. Implications of artificial intelligence for a user-defined technical information system. Applied Science Associates, Inc., 1983.
- Merrill, M.D. & Tennyson, R.D. Teaching concepts: An instructional design guide. Englewood Cliffs, New Jersey: Educational Technology Publications, 1977.
- Miller, L.A. Project EPISTLE: A System for the automatic analysis of business correspondence. Behavioral Sciences and Linguistics Group, IBM Thomas J. Watson Research Center, 19-21, 1980.
- Miller, L.A., Heidorn, G.E., Jensen, K. Text-critiquing with the EPISTLE system: An Author's aid to better syntax. IBM Thomas J. Watson Research Center, 1981.
- Montague, W.E., & Wulfek II, W.H. Computer-based Instruction: Will it Improve Instructional Quality? Training Technology Journal, 4-17, 1984.

- Montague, W.E., Ellis, J.A., & Wulfeck II, W.H. After years of instructional research do we know more than grandma did about how to teach people. Invited Address, Div. C. Section 4, American Educational Research Association, Session 23.02. Eric Document, 1983.
- Montague, W.E., & Wulfeck II, W.H. Improving the quality of Navy training: The Role of R&D in support of instructional systems design. San Diego, CA: Navy Personnel Research and Development Center. Special Report 82-19, 1982.
- Norman, D.A. Stages and levels in human-machine interaction. ICS Report #8402, UCSD HMI Project. 1984.
- Olsen, J.B., Bunderson, C.V., Baillio, B. Comparative Evaluation of a Prototype Intelligent Videodisc system. WICAT Institute for Education.
- O'Neil, H.F., Jr. (Ed.). Procedures for instructional systems development. New York: Academic Press, 1979.
- O'Neil, H.F., Jr. (Ed.). Issues in instructional systems development. New York: Academic Press, 1979.
- O'Neil, H.F. (Ed.). Computer-based instruction: A state-of-the-art assessment. New York: Academic Press, 1981.
- Reisner, P. Human factors studies of database query languages: A survey and assessment. Computing Surveys, 13, 13-31, 1981.
- Richardson, J.J. On-line task analysis in maintenance simulation. In Proceedings of the 5th Interservice/Industry Training Equipment Conference, Washington, DC, November 14-16, 1983. (Also to appear in Machine-Mediated Learning, 2(1), 1984.)
- Richardson, J.J. An integrated design and development system for graphics simulation. In Computer Literacy, Intelligent CAI, 1982 Conference Proceedings of the Association for the Development of Computer-Based Instructional Systems (ADCIS), Bellingham, WA, 1982.
- Riley, M., & O'Malley, C. Planning nets: A framework for analyzing user-computer interactions. ICS Report #8402, UCSD HMI Project, 1984.
- Roberts, G., McCracken, D., Newell, A. The ZOG approach to man-machine communication. Technical Report CMU-CS-79-148, Department of Computer Science, CMU, 1979.
- Robertson, S.P. Goal, plan, and outcome tracking in computer text-editing performance. Technical Report #25, Cognitive Science Program, Yale University, 1983.
- Steffen, D.A., & Kottenstette, J.P. Training and Performance Support System: A New Dimension in Individualized Instruction. DOD Ninth Psychology Symposium, 1984.

- Vaiana, M.E., Shapiro, N., LaCasse, M. Clarify: An on-line guide for revising technical prose. Technical Report #N-2037-RC, Rand, Santa Monica, 1983.
- Waldrogs, H. Composing prose with Wandah. PC Magazine, 161-164, 1984.
- Walker, D.F., & Hess, R.D., (Eds.). Instructional software: Principles and perspectives for design and use. Belmont, CA. Wadworth Inc. 1984.
- Williams, M.D., Tous, F.N., Fikes, R.E., Henderson, A., & Malone, T. RABBIT: Cognitive science in interface design. Proceedings of the Fourth Annual Cognitive Science Conference, Ann Arbor, MI, 1982.
- Williges, B.H., & Williges, R.C. User Considerations in Computer-Based Information Systems. Virginia Polytechnical Institute and State University, 1981.
- Wilson, L.S. Presenting TICCIT: State-of-the-Art Computer-Based Instruction. National Security Industrial Association, 1984.

APPENDIX

Structuring Questions for Small Group Discussions

(Sessions VI, VII, VIII)

1. For what functions, e.g., analysis, instructional design, are automated authoring aid technologies likely to have the largest short-term (3-5 years) impact?
2. What functions (because of their success) no longer require additional R&D effort?
3. What functions should be discarded temporarily (because of lack of knowledge) or permanently (because of lack of progress or potential impact)?
4. What functions or areas suggest the need for relatively large investment for a probable short-term (3-5 years) impact?
5. What should be criteria for the assessment of internal features and external effects of AAAS?
6. For what learning tasks are AAAS best suited?
7. What new training requirements for AAA implementation should be developed?
8. How extensible are extant designs for AAAS to a wide range of tasks and delivery conditions? What can be improved rapidly?

AGENDA

AUTOMATED AUTHOR AIDING SYSTEMS CONFERENCE (AASC)
UCLA CENTER FOR THE STUDY OF EVALUATION
JUNE 6 AND 7, 1984
FACULTY CENTER, UCLA
California Room
Hilgard and Westholme Avenues

WEDNESDAY, JUNE 6, 1984

8:00 - 8:30	Coffee and Registration	
8:30 - 8:50	Opening Statement	H. O'Neil, Jr. Army Research Institute
8:50 - 9:00	Welcome and Introductions	E. Baker UCLA
9:00 - 10:30	Session I: Computer-Based Instruction: Will It Improve Instructional Quality?	
	Chairman and Commentator:	W. Dick Florida State University
	Presenters:	W. Montague & W. Wulfeck NPRDC
10:30 - 10:45	Coffee Break	
10:45 - 12:15	Session II: General Issues in the Design of Automated Authoring Aids	
	Chairman and Commentator:	M. D. Merrill University of Southern California
	Presenters:	M. Marken Northrop Corp. G. Roid Western Psychological Services M. Watson McDonnell Douglas Corp.
12:30 - 1:30	LUNCH	(Sequoia Dining Room)

1:30 -3:00 Session III: Automated Aids for Instructional Design

Chairman and Commentator: R. Davis
University of Illinois

Presenters: S. Goldberg
ARI/Fort Knox

S. Taylor
Control Data Corp.

R. Bauer
ACTO

R. Akscyn
Carnegie-Mellon University

S. Morris
Texas Instruments

3:00 - 3:15 Break

3:15 - 5:00 Session IV: Issues in Authoring Aids for Document Design

Chairman and Commentator: T. Duffy
Carnegie-Mellon University

Presenters: D. Braby, Naval Training
Equipment Center

M. Friedman, UCLA

R. Glushko
Bell Laboratories

R. Gorsuch
Fuller Theological Seminary

S. Keenan
Bell Laboratories

J. Bean
Hughes Aircraft

5:00 - 5:10 Housekeeping

5:10 - 6:00 System Demonstrations (Sierra Room)

5:30 - 7:30 Reception

7:30 - 9:30 Dinner

THURSDAY, JUNE 7, 1984

8:00 - 8:30 Coffee

8:00 - 9:00 System Demonstrations (Sierra Room)

9:00 - 10:30 Session V: Applications of Artificial Intelligence to
Automated Authoring

Chairman and Commentator: W. Feurzeig
Bolt, Beranek & Newman, Inc.

Presenters: M. Jensen, Boeing Aerospace
Company

W. Stacy, Scientific
Systems, Inc.

R. Spiro
University of Illinois

A. Gibbons
WICAT Systems

10:30 - 10:45 Coffee Break

10:45 - 12:15 Session VI: Small Group Discussions

12:30 - 1:30 LUNCH (Si-quoia Dining Room)

1:30 - 2:00 Session VII: Small Group Synthesis

2:00 - 3:00 Session VIII: Small Group Presentations

Chairman and Commentator: D. Fletcher
University of Oregon

3:00 - 3:30 Conference Summary D. Fletcher
University of Oregon

3:30 Adjournment of Group Meeting

3:30 - 5:00 Future Directions of Automated Authoring Systems
Closed Session: Government Personnel only
(Moore Hall, Room 120)

Chairman: H. O'Neil, Jr

Automated Author-aiding Systems Conference
June 6-7, 1984

Dr. Robert Akscyn
Carnegie Mellon University
4628 Henry Street
Pittsburgh, PA 15213

Dr. Nancy Atwood
Center for the Study of Evaluation
UCLA Graduate School of Education
Los Angeles, CA 90024

Dr. Eva L. Baker
Director
Center for the Study of Evaluation
UCLA Graduate School of Education
Los Angeles, CA 90024

Mr. Robert Bauer
U.S. Army Training Support Center
ATTN: DRSEL-FED
Fort Eustis, VA 23604

Mr. John Bean
Hughes
Bldg. A-11, Mail Stop 28917
P.O. Box 9399
Long Beach, CA 90810

Mr. Douglas Bernon
Hazeltime
7680 Old Springhouse Road
McLean, VA 22102

Colonel Gary Bloedon
Perceptronics
6271 Varie Avenue
Woodland Hills, CA 91367

Mr. Dick Braby
Naval Training Equipment Center
ATTN: TAEG, Code N-1
Orlando, FL 32813

Mr. James Cochran
Commander
U.S. Army Communication-Electronics Command
Attn: DRSEL-ME-PMP
Fort Mammouth, NJ 07703

Mr. Bill Creel
U.S. Army Missile Command
ATTN: DRSMI-SNP
Redstone Arsenal, AL 35898

Mr. Tony Cuneo
U.S. Army Tank-Automated Command
ATTN: DRSTA-MPB
Warren, MI 48092

Dr. Pat Cwynar
Denver Research Institute
University of Denver
Denver, CO

Dr. Robert Davis
Professor of Education
University of Illinois
Champaign, IL 61820

Dr. Walter Dick
Florida State University
Room 307, Stone Building
Tallahassee, FL 32306

Dr. William Doherty
Center for the Study of Evaluation
UCLA Graduate School of Education
University of California
Los Angeles, CA 90024

Dr. Thomas M. Duffy
Communications Design Center
Carnegie Mellon University
Pittsburgh, PA 15213

Dr. Jean L. Dyer
Cmndt, USAIS (US Army Infantry School)
ATSH-I-A
Fort Benning, GA 31905

Dr. Michael G. Dyer
Computer Science Department
3532 Boelter Hall
University of California - Los Angeles
Los Angeles, CA 90024

Mr. Edward Earl
Management Analyst
2461 Eisenhower Avenue
(DAAG)-PAZ-E
Alexandria, VA 22331

Ms. Barbara Eden
Air Force Human Resource Laboratory
ATTN: AFHRL-ID
Lowry AFB, CO 80230

Mr. Walt Eichwald
Jet Propulsion Laboratory
Information Systems Engineering
California Institute of Technology
4800 Oak Grove Drive
Pasadena, CA 91109

Dr. Thomas Elliott
Applied Science Associates, Inc.
P.O. Box 158
Valencia, PA 16059

Mr. Phillip Ender
Center for the Study of Evaluation
UCLA Graduate School of Education
Los Angeles, CA 90024

Mr. Richard Feifer
Center for the Study of Evaluation
UCLA Graduate School of Education
Los Angeles, CA 90024

Dr. Robert Feuge
Mathetics
P.O. Box 2665
San Diego, CA 92126

Dr. Walter Feurzeig
Bolt, Baranek and Newman, Inc
Information Sciences Division
Ten Moulton Street
Cambridge, MA 02238

Dr. J. Dexter Fletcher
University of Oregon
College of Education
Eugene, OR 97403

Dr. Morton Friedman
Director
Evaluation of Instruction Program
University of California
Los Angeles, CA 90024

Mr. Andy Gibbons
Director
Courseware Development
Training Systems Division
WICAT, P.O. Box 539
Orem, UT 84054

Dr. Ralph Gilstrap
Director
Government Marketing Support
WICAT, P.O. Box 539
Orem, UT 84054

Mr. Robert J. Glushko
Bell Laboratories
1B217
Whippany, NJ 07981

Dr. Steven Goldberg
ARI Field Unit-Fort Knox
Steel Hall
Fort Knox, KY 40121

Dr. Dwight Goehring
ARI/U.S. Army Research Institute
Presidio/Monterey P.O. Box 5787
Presidio of Monterey, CA 93944

Ms. Mary Gordon
Center for the Study of Evaluation
UCLA Graduate School of Education
Los Angeles, CA 90024

Dr. Richard L. Gorsuch
Graduate School of Psychology
Fuller Theological Seminary
177 N. Madison
Pasadena, CA 91101

Dr. Joan Herman
Center for the Study of Evaluation
UCLA Graduate School of Education
Los Angeles, CA 90024

Dr. Jack Hiller
U.S. Army Research Institute Office
Presidio/Monterey Field Unit
P.O. Box 5787
Presidio of Monterey, CA 93944

Dr. Robert Jacobs
Vice President and Director
Perceptronic
Training and Simulation Group
6271 Varie! Ave.
Woodland Hills, CA 91367

Dr. Marty Jensen
Boeing Aerospace Company
P.O. Box 3999
Seattle, WA 98124
Mail Stop 8777

Ms. Wanetta Jones
Center for the Study of Evaluation
UCLA Graduate School of Education
Los Angeles, CA 90024

Dr. Otto Kahn
ARI/U.S. Army Research Institute
Presidio/Monterey P.O. Box 5787
Presidio of Monterey, CA 93044

Dr. Joseph H. Kanner
Training Technology Agency
ATTN: ATTG-D
Fort Monroe, VA 23651

Ms. Stacey Keenan
Bell Laboratories
190 River Road, Room S-315
Summit, NJ 07901

Mr. Anthony Keller
U.S. Army Communication-Electronics Command
Attn: DRSEL-ME-PMP
Fort Monmouth, NJ 07703

Mr. Robert Lake
Hughes Support Systems
Bldg A-1
Mail Stop 3A909
P.O. Box 9399
Long Beach, CA 90810

Lt. Kevin Leak
ATTN: ATAM-IT
Fort Monroe, VA 23651

Ms. Aeri Lee
Center for the Study of Evaluation
UCLA Graduate School of Education
Los Angeles, CA 90024

Mr. Robert Logan
Systran Incorporation
70 W. Hubbard
Chicago, IL 60610

Dr. Kenneth Majer
President
Information-Decisions
424 F Street
San Diego, CA 92101

Dr. Mara Marken
Instructional Designer
Training Plans
Northrop Corporation
Aircraft Division
One Northrop Avenue
Hawthorne, CA 90250

Dr. David McArthur
Center for the Study of Evaluation
UCLA Graduate School of Education
Los Angeles, CA 90024

Dr. David Merrill
801 Waite Phillips Hall
University Park
Los Angeles, CA 90007

Mr. Steven Morris
Project Manager
Texas Instruments
13510 North Central Expressway
Mail Stop 394
ATTN: John Harkin
Dallas, TX 75265

Dr. William Montague
Navy Personnel Research and Development Center
Code 05
San Diego, CA 92152

Dr. Harold O'Neil, Jr
Army Research Institute
5001 Eisenhower Avenue
Alexandria, VA 22333

Dr. Ray Perez
Army Research Institute
5001 Eisenhower Avenue
Alexandria, VA 22333

Mr. Lee Rivers
Hughes Support Systems
Bldg A-1
Mail Stop 3A909
P.O. Box 9399
Long Beach, CA 90810

Mr. Ron Renfro
Mathetics Inc.
10721 Treena Street
Suite 217
San Diego, CA 92131

Master Sergeant Lloyd M. Robbins
USAF OMC/OMT
Randolph AFB, TX 78150

Dr. Gale Roid
Western Psychological Services
12031 Wilshire Boulevard
Los Angeles, CA 90025

Dr. Thomas Roth
Applied Science Associates, Inc.
P.O. Box 158
Valencia, PA 16059

Dr. Michael Samet
Perceptronic
6271 Varie Avenue
Woodland Hills, CA 91367

Dr. Richard Shavelson
UCLA Graduate School of Education
Los Angeles, CA 90024

Dr. Al Silliman
Jet Propulsion Laboratory
Information Systems Engineering
California Institute of Technology
4800 Oak Grove Drive
Pasadena, CA 91109

Dr. Rand Spiro
Center for the Study of Reading
51 Gerty Drive
Champaign, IL 61820

Mr. Webb Stacey
Scientific Systems, Inc.
54 Rindge Avenue Ext.
Cambridge, MA 02140

Dr. Cathleen Stasz
The RAND Corporation
1700 Main Street
Santa Monica, CA 90406

Mr. William Stembler c/o Don Kimberline
U.S. Army Training Support Center
ATTN: DRSEL-FED
Fort Eustis, VA 23604

Ms. Susan Taylor
Manager Courseware Development
Control Data
6003 Executive Boulevard
Rockville, MD 20852

Dr. Marsha Watson
McDonnell Douglas (35-36)
3855 Lakewood Boulevard
Long Beach, CA 90846

Dr. Lois Wilson
Hazel'tine
7680 Old Springhouse Road
McLean, VA 22102

Dr. Merlin Wittrock
UCLA Graduate School of Education
University of California
Los Angeles, CA 90024

Dr. Jack Wojnaroski
Jet Propulsion Laboratory
Information Systems Engineering
California Institute of Technology
4808 Oak Grove Drive
Pasadena, CA 91109

Ms. Pauline Wojnaroski
Jet Propulsion Laboratory
Information Systems Engineering
California Institute of Technology
4808 Oak Grove Drive
Pasadena, CA 91109

Dr. Larry Wood
Department of Psychology
SWKT Box 75
Brigham Young University
Provo, UT 84604

Dr. Wallace Wulfeck
Navy Personnel Research and Development Center
Code 51
San Diego, CA 92152

INFORMATION BRIEF FOR AUTOMATED AUTHORIZING AIDS CONFERENCE

TOPIC: OVERVIEW AND INTRODUCTION

JUNE 1984

PRESENTER: HAROLD F. O'NEIL, JR.
DIRECTOR, TRAINING
RESEARCH LABORATORY

TELEPHONE: AV 284-8622
202 274-8622

PROBLEM

- COURSEWARE BOTTLENECK IN CAI SYSTEMS
 - - NO EFFICIENT WAY TO CREATE LARGE AMOUNTS OF CAI MATERIALS
- ISD FAILURE

ISD FAILURE

- TOOLS NOT AVAILABLE
- LARGE SCALE IMPLEMENTATION
- RESOURCES INADEQUATE
- LITTLE TRAINING
- TURBULENCE OF KEY PERSONNEL
- LACK OF SYNTHESIS/INTEGRATION

OBJECTIVE: STATE OF ART ASSESSMENT

- WHAT DO WE KNOW?
- WHAT SHOULD WE DO?
 - ONLINE AIDS TO CREATE
ONLINE INSTRUCTION
 - EMPHASIS ON DEVELOPMENT
NOT RESEARCH
 - EMPHASIS ON "HOW TO"

EVENTUAL PRODUCTS:

- AN AUTOMATED AUTHORIZING SYSTEM
- RESEARCH AGENDA
 - AI EXPERT SYSTEM FOR MEDIA SELECTION
- REVISION OF 350-7

APPROACH: STATE OF THE ART ASSESSMENT

- CONDUCT CONFERENCE
 - GOVERNMENT, INDUSTRY, ACADEMIA
- ADDITIONAL EXPERTISE VIA CONSULTANTS
- REPORT COMPLETED AUGUST 1, 1984

FRAMEWORK

MODEL: SYSTEMS APPROACH TO TRAINING - TRADOC REGULATION No. 350-7

DETAIL PROCEDURES: TRADOC PAM 350-30

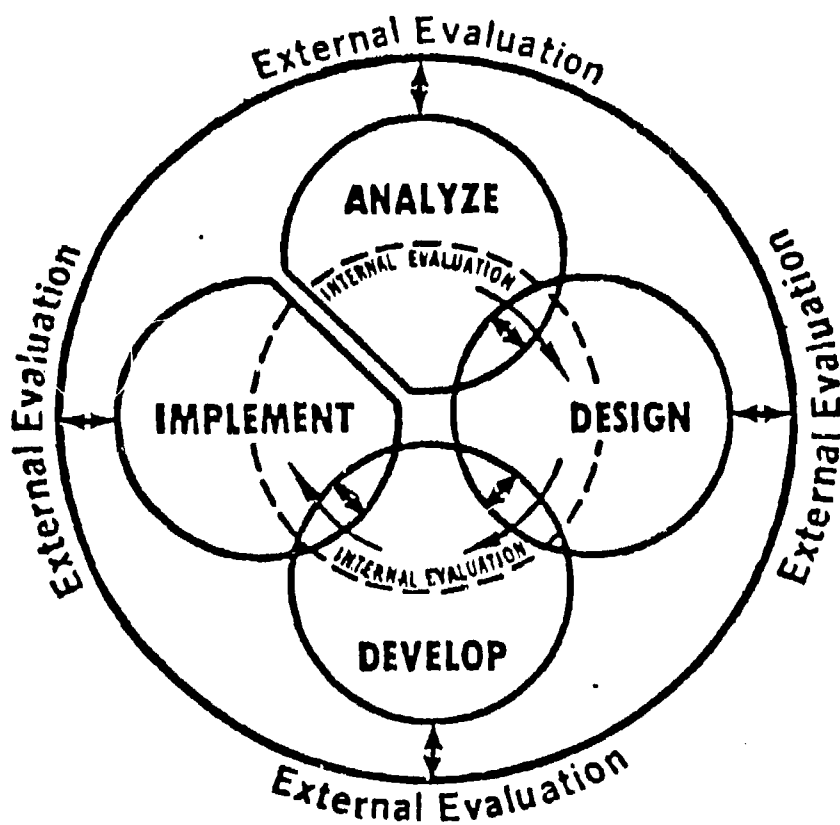
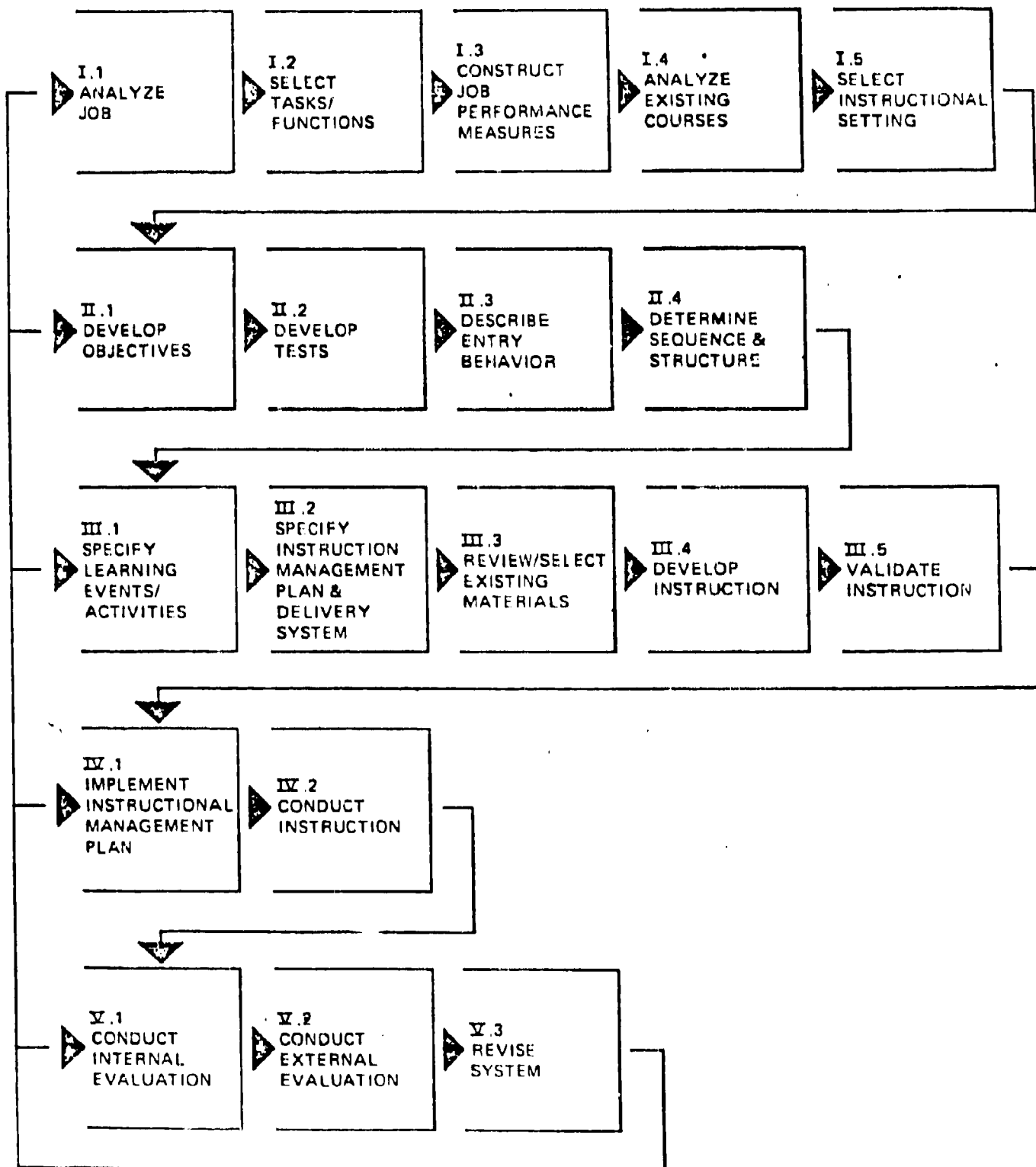


Figure 2-2. A model for a systems approach to training.

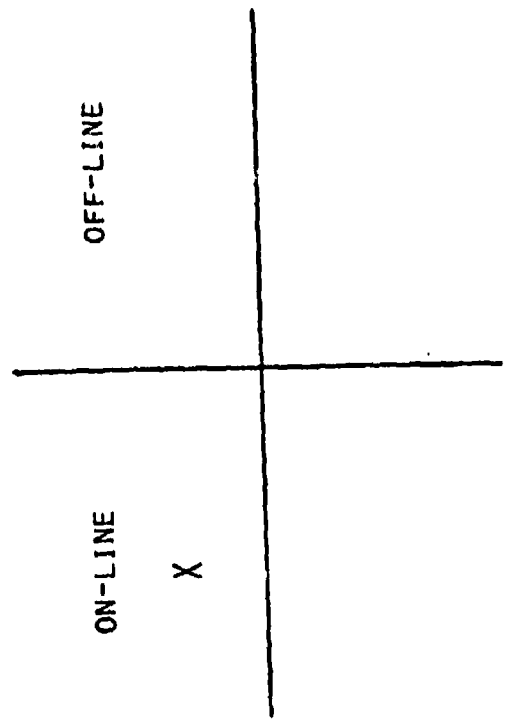
THE BLOCKS IN EACH PHASE ARE:



WHAT I DON'T WANT

- DISCUSSION OF "IDEAL" MODEL
 - ARMY ALREADY HAS ONE - 350-7
- IMPACT OF COGNITIVE SCIENCE
 - TOPIC OF PRIOR CONFERENCE BY NPRDC/ARI
- CRITERIA FOR EVALUATION OF AID'S
 - CSE TASK
- AUTOMATED AIDs FOR OFFLINE MATERIALS
 - GET OFF PAPER BASE
- AI APPROACH
 - FOCUS ON DEVELOPMENT, NOT RESEARCH
- AIDs FOR VIDEO DISC
 - ACTO HAS ONGOING CONTRACT

TARGET INSTRUCTION



ON-LINE

OFF-LINE

TYPE OF
AID